PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04N 1/32

A1

(11) International Publication Number: WO 95/22865

(43) International Publication Date: 24 August 1995 (24.08.95)

(21) International Application Number: PCT/US95/02120

(22) International Filing Date: 16 February 1995 (16.02.95)

(30) Priority Data:
08/198,364 18 February 1994 (18.02.94) US

(71) Applicant: MICROSOFT CORPORATION [US/US]; One Microsoft Way, Redmond, WA 98052-6399 (US).

(72) Inventors: MATHUR, Sharad; 14924 Northeast 76th Court, Redmond, WA 98052 (US). MENEZES, Arul; 13930 Northeast 13th Street, No. 6-2, Bellevue, WA 98005 (US). VANBUSKIRK, Michael, Robert; 9604 172nd Avenue Northeast, Redmond, WA 98052 (US). GALLO, Kevin, Thomas; 4850 156th Avenue Northeast, No. 267, Redmond, WA 98052 (US).

(74) Agents: KIKLIS, Michael, L. et al.; Seed and Berry, 6300 Columbia Center, 701 Fifth Avenue, Seattle, WA 98104-7092 (US).

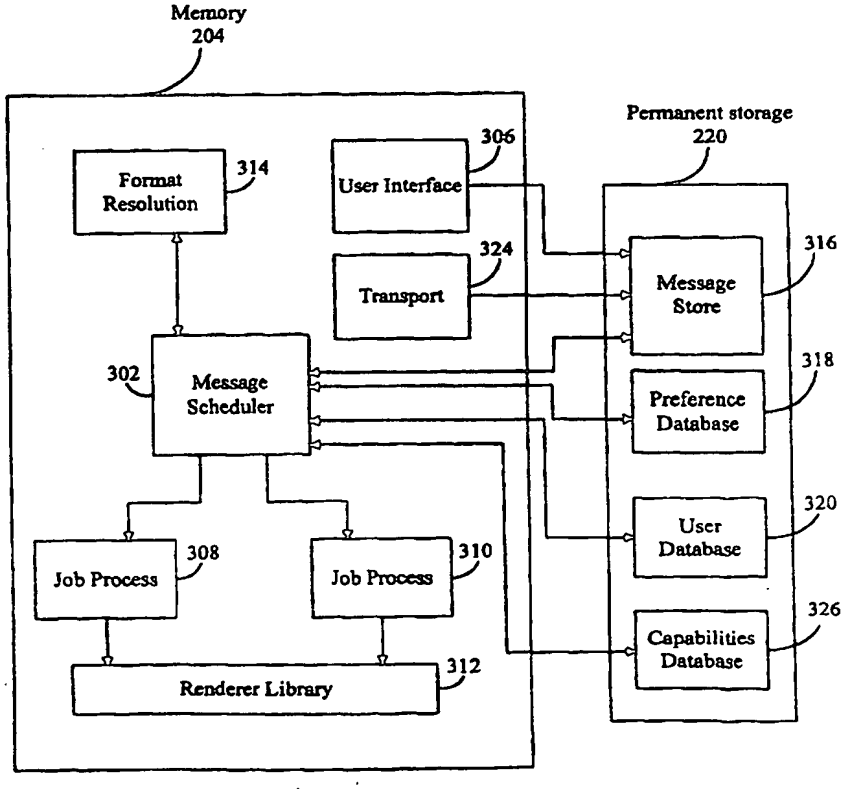
(81) Designated States: JP, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(54) Title: A SYSTEM FOR TRANSFERRING MESSAGES BETWEEN INPUT AND OUTPUT DEVICES IN A COMMUNICATION DEVICE

(57) Abstract

A method and system for processing messages is provided. In a preferred embodiment of the present invention, a message scheduler, a job process, and a format resolution module are used. The message scheduler identifies when a new message is to be processed, schedules the new message for processing, invokes the format resolution module to determine a method of converting data in the message to an appropriate format for a destination device or file, and invokes a job process. The format resolution module determines an appropriate method for converting data in the message from a format acceptable to the source devices or files to a format acceptable to the destination devices or files and creates a target message containing a reference to a destination device or file. The job process sends the data between the source devices or files to a destination device or file and performs the appropriate conversion. The preferred embodiment of the present invention is designed to allow for the support of new devices and new data formats with little change to the existing system through the use of a building block methodology. Therefore, new devices and data formats can be supported in an easy and efficient manner.



FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria
AU	Australia
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CS	Czechoslovakia
CZ	Czech Republic
DE	Germany
DK	Denmark
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgystan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakhstan
LI	Liechtenstein
LK	Sri Lanka
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SI	Slovenia
SK	Slovakia
SN	Senegal
TD	Chad
TG	Togo
TJ	Tajikistan
TT	Trinidad and Tobago
UA	Ukraine
US	United States of America
UZ	Uzbekistan
VN	Viet Nam

Description

A SYSTEM FOR TRANSFERRING MESSAGES BETWEEN INPUT AND OUTPUT DEVICES IN A COMMUNICATION DEVICE

Technical Field

The present invention relates generally to communication devices and, more particularly, to the transferring of messages by a communication device.

Background of the Invention

Facsimile machines allow users to send and/or receive documents via telephone lines. Figure 1 depicts a conventional facsimile machine 100. The facsimile machine 100 is connected to a telephone line 110. The facsimile machine 100 contains a modem 106, a memory 104, a central processing unit (CPU) 102, a printer 108, and a scanner 112. The modem 106 is used for converting received messages from the telephone line 110 from an analog format into a digital format. The modem 106 also converts data received from the CPU 102 from a digital format to an analog format for transmission across the telephone line 110. When a message is sent, the CPU 102 is responsible for receiving digital data from the memory 104, compressing the digital data and sending the digital data to the modem 106. In addition, the CPU 102 is responsible for receiving digital data from the modem 106, sending the digital data to the memory 104, and decompressing the digital data when a message is received. The memory 104 is used for storing digital data while the digital data is being sent or received. The printer 108 is an output device responsible for receiving digital data from the CPU 102 and for printing the digital data. The scanner 112 is an input device responsible for receiving a document, digitizing the document, and sending the digitized data to the CPU 102.

Conventional facsimile machines are a type of communication device that processes messages in a very limited fashion. Conventional facsimile machines perform the function of receiving messages from a telephone line and printing the received messages, as well as scanning documents and transmitting the scanned data across telephone lines. Conventional facsimile machines do not use other devices than a modem, printer and scanner nor can conventional facsimile machines delay the processing of messages.

Summary of the Invention

In accordance with a first aspect of the present invention, a method is executed in a communication device having a number of devices and files. In accordance with this method of the first aspect of the present invention, at least one device or file is designated as the message source and at least one device or file is designated as the message sink. Further, in accordance with this method of the first aspect of the present invention, the communication device contains a message store for storing messages, a job process for sending data in a message from the message source to the message sink, a renderer library and a format resolution module. This method processes messages by first viewing the message store. After viewing the message store, this method schedules all new messages contained in the message store. When a message in the message store is a target message, this method spawns a job process to process the target message. When a message in the message store is a source message, this method determines the message sink for the source message and invokes the format resolution module.

In accordance with a second aspect of the present invention, the communication device includes a message scheduler, a job process, and a format resolution module. The message scheduler provides for invoking job processes for each message and for invoking the format resolution module for each message. The job process provides for sending data from a message source to a message sink and for converting the data. The format resolution module provides for creating a topology list of renderers for converting the data from a format acceptable to the message source to a format acceptable to the message sink.

Brief Description of the Drawings

Figure 1 depicts a block diagram of a conventional facsimile machine.

Figure 2 depicts a block diagram of a communication device suitable for practicing the preferred embodiment of the present invention.

Figure 3 depicts a more detailed block diagram of the memory and the permanent storage of a communication device suitable for practicing the preferred embodiment of the present invention.

Figure 4 depicts a more detailed block diagram of the message store.

Figure 5 depicts a more detailed diagram of a message in the message store.

Figure 6 depicts a flow chart of the steps performed by the preferred embodiment for a send function.

Figure 7 depicts a flow chart of the steps performed by the preferred embodiment for a receive function.

Figures 8A and 8B depict a flow chart of the steps performed by the message scheduler.

Figure 9 depicts a flow chart of the steps performed by the format resolution.

Figure 10 depicts a flow chart of the steps performed by the job process.

Detailed Description of the Invention

A preferred embodiment of the present invention provides for an improved communication device for processing messages that provides more functionality than a conventional facsimile machine. The preferred embodiment of the present invention provides the ability to transmit and receive messages from many different types of devices or files. Moreover, the preferred embodiment can transmit a message received from many devices or files to many devices or files. In addition, the preferred embodiment of the present invention converts data received from a device or file into a format acceptable to another device or file. As a result, data, such as documents, may be readily directed to different Input and Output ("I/O") devices. The preferred embodiment of the present invention is designed to allow for the support of new devices and new data formats with little change to the existing system through the use of a building block methodology. Therefore, new devices and data formats can be supported in an easy and efficient manner.

Although the preferred embodiment of the present invention is described relative to a facsimile machine, one skilled in the art will recognize that the present invention can be practiced with other I/O devices or communication devices.

The preferred embodiment of the present invention provides three functions in a facsimile machine which will be discussed in more detail below: a send function, a receive function, and a copy function. The send function allows a user to send data from one or more devices or files ("message sources") to one or more other devices or files ("message sinks"). The copy function allows a user to copy data from the scanner to the printer. The receive function allows the user to receive data from the modem and send the data to a preferred device of the user. The receive function is performed automatically by the preferred embodiment without user intervention. Although the preferred embodiment of the present invention is described as providing three functions, one skilled in the art will recognize that the present invention may provide other functions.

Figure 2 depicts a block diagram of a facsimile machine suitable for practicing the preferred embodiment of the present invention. The facsimile machine 200 is connected to a telephone line 210, a network 222, a parallel device 224 and a serial device 226. The facsimile machine 200 contains a memory 204, a central processing unit (CPU) 202, a keyboard 228, a printer 208, a scanner 212, a network interface 216, a modem 206, a serial port 214, a parallel port 218, and a permanent storage device 220. Although the preferred embodiment is described as containing one of each type of device, one skilled in the art will recognize that the preferred embodiment may be used with more than one of each type of device. The memory 204 is used as a storage area for messages as the messages are processed in the facsimile machine 200. The memory 204 can be any of a number of different types of memory, including, but not limited to, dynamic random access memory and static random access memory. The CPU 202 processes messages contained in the memory 204 and is responsible for sending and receiving data from the permanent storage 220, the keyboard 228, the parallel port 218, the serial port 214, the network 216, the modem 206, the scanner 212, and the printer 208.

The keyboard 228 is an input device that allows the user to input information and commands into the facsimile machine 200 for processing by the CPU 202. The printer 208 is an output device that is responsible for receiving data from the CPU 202 and for printing the received data. The scanner 212 is an input device that accepts documents, digitizes the data contained in the document into a digital format, and sends the digital data to the CPU 202. The network interface 216 acts as an entry point to a network 222. The network interface 216 is responsible for receiving data from other devices on the network 222 and sending the received data to the CPU 202, and for receiving data from the CPU 202 and sending the received data to other devices on the network 222. The network 222 can be any of a number of different networks including, but not limited to, local area networks and wide area networks. The modem 206 is responsible for receiving digital data from the CPU 202 and for converting the digital data into an analog format for transmission across the telephone line 210. In addition, the modem 206 is responsible for receiving analog data from the telephone line 210, for converting the analog data into a digital format, and for sending the digital data to the CPU 202. The serial port 214 is an access point to other devices that communicate in a serial manner (i.e., a serial device 226). The term "Serial" refers to the transmission of data in a bit by bit sequential manner as opposed to parallel transmission wherein all the bits of a group of data are sent simultaneously. The types of devices that can be connected to a serial port are other computers or peripheral

devices. The parallel port 218 acts as an access point to devices that communicate in a parallel manner (i.e., a parallel device 224). An example of a parallel device is a parallel printer. A parallel printer, for example, instead of receiving data in a serial manner (bit by bit), receives one byte of data (8 bits) at a time. The permanent storage 220 is a non-volatile storage device for storing data in between invocations of the facsimile machine 200. An example of a permanent storage device 220 is a hard disk.

Although the preferred embodiment of the present invention is described as being practiced on a facsimile machine like that shown in Figure 2, one skilled in the art will recognize that the present invention may be practiced with other facsimile machine configurations, and the facsimile machine 200 may interface with other components.

Figure 3 depicts a more detailed block diagram of the memory 204 and the permanent storage 220 of the facsimile machine 200 and illustrates the interactions between the depicted components. The memory 204 contains a user interface process 306, a transport process 324, a message scheduler 302, a format resolution module 314, a renderer library 312, and two job processes 308,310. The permanent storage 220 contains a preferences database 318, a user database 320, a capabilities database 326, and a message store 316. The user interface process 306 is responsible for receiving commands and information from a computer user through the keyboard 228. In response to the received commands and information from the keyboard 228, the user interface process 306 creates messages and stores the created messages into the message store 316. Two commands in particular that the user interface 306 allows the user to input are the send command and the copy command. The transport process 324 is responsible for receiving data from the modem 206 or other device, creating a message from the received data, and then storing the message into the message store 316. The message scheduler 302 is responsible for viewing the message store 316. Upon viewing the message store 316, the message scheduler 302 identifies new messages, processes the new messages, and updates the status of the messages in the message store 316. The format resolution module 314 is a dynamically linked library module that is responsible for determining a method for converting a message from a format acceptable to a source device or file to a format acceptable to a destination device or file and for formatting the message according to the preferences of the computer user. The source devices or files are referred to as the "Message Sources" and the destination devices or files are referred to as the "Message Sinks". A "dynamically linked library module" is a shared library module that is linked at run time. Although

some components of the preferred embodiment are described as dynamically linked libraries, one skilled in the art will recognize that the dynamically linked libraries can be statically linked libraries, processes, functions, modules, or other software entities. The format resolution module 314 determines a method for converting data in a message by creating a topology list of renderers. A "renderer" is a function in the renderer library 312 that is responsible for converting data from one format into another. A "topology list of renderers" refers to a group of renderers in a specific order in which each renderer is to be executed for conversion of the data from a format acceptable to the message source to a format acceptable to the message sink. The job process 308, 310 is responsible for invoking renderers from the renderer library 312 in the order that the topology list dictates and for sending data between the message source and the message sink.

The preferences database 318 maintains the preferences of each user of the facsimile machine 200 for a preferred message sink for a received message as well as formatting preferences for a message. In using the preferences database 318, the destination user name is used as an index and an identifier of the preferred message sink and formatting preferences for that user name are returned. "Formatting" refers to additional information added to a message that does not vary the content of the message. The term "formatting" includes overlays which are stylistic information added to a message much like headers or footers are added to a document. Also, the term "formatting" includes branding which refers to adding time and date information to a message when a message is printed. Therefore, the preferences database 318 contains user preferences for a preferred message sink as well as formatting preferences. The user database 320 is used to determine the address and message sink for a specific user. In using the user database 320, the destination user name is used as an index and an address for the destination user is returned with an identifier of a message sink. The capabilities database 326 is used to determine the data format acceptable to a message sink. The capabilities database 326 has an entry for each device in the facsimile machine 200 and the data formats (capabilities) acceptable to the devices. The message store 316 acts as the repository for all messages in the facsimile machine 200.

Figure 4 depicts a more detailed block diagram of the message store 316. The message store 316 is a storage unit for messages in the facsimile machine 200. The message store 316 as depicted contains six messages 402, 404, 406, 408, 410, and 412. Figure 5 depicts a more detailed diagram of message 402. Although message 402 is depicted, the other messages contained in the message store 316 have the same

structure. Message 402 contains a header portion 502 and an attachments portion 504. The header portion 502 contains a status field 506, a type field 508, a delivery time field 510, a message source field 512, and a destination user name field 514. The attachments portion 504 contains each attachment of the message 514, 516. The status field 506 indicates the stage of processing of the message. A message can be a new message, a schedulable message, or a completed message. A new message is a message that has been deposited into the message store 316 by the user interface process 306 or the transport process 324, but has not yet been processed by the message scheduler 302. A schedulable message in the message store 316 is a message where the time for delivery of the message has been reached. A completed message in the message store is a message where transmission of the referenced data has been completed. The type field 508 is used to indicate whether the message is a source message or a target message. A source message is a message that contains references (e.g., pipes and filenames) to the source data that is to be sent. A target message is a message that refers to the message sink. A "pipe" is a reference to a communication channel in memory where data is transferred to and from a device. The delivery time field 510 is used to indicate the date and time for which delivery is to occur. A message is not schedulable until the date and time in the delivery time field 510 has been reached. The message source field 512 contains an identifier of the source devices or files from which the data is to be received (i.e., the message source). An attachment 514, 516 to a message is a reference to data that is a filename, a pipe, or a direct reference to a device. A message may contain many attachments. Although the message 402 may contain many attachments, the data referenced in the message is considered by the preferred embodiment to be one message.

To illustrate the interactions of the components of the preferred embodiment, consider the following example of the send function and the receive function. Figure 6 depicts a flow chart of the steps performed by the preferred embodiment during the send function. In this example, a remote user initiates the send function. First, the remote user inputs commands and information to a remote personal computer ("PC") (step 602). The remote user inputs information such as the date and time for delivery, the source device or file, and the destination user name. Then the PC sends the information to the facsimile machine 200, wherein the facsimile machine 200 receives the information through the serial port 214 and sends the information to the transport process 324 (step 603). The transport process 324 next creates a message from the received user information, marks the created message as being new, and stores the new message into the message store 316 (step 604). Upon finding a new message in

the message store 316, the message scheduler 302 determines the message sink for the message by accessing the user database 320 with the destination user name 514 (step 606). The message scheduler 302 accesses the user database 320 with the destination user name 514 and the user database 320 returns the address and message sink of the destination user. The message scheduler 302 then invokes the format resolution module 314 to create a topology list of renderers and to create a target message for each destination (message sink) of the message (step 608). The format resolution module 314 determines the capabilities of each message sink by accessing the capabilities database 326 and determines the formatting preferences of the computer user by accessing the preferences database 318 (step 610). The format resolution module 314 then determines a topology list of renderers to convert each attachment in the message from a format acceptable to the message source to a format acceptable to each message sink (step 612). The format resolution next creates a target message for each message sink (step 614). After receiving the results from the format resolution module 314, the message scheduler 302 spawns one job process 308, 310 for each message sink to send data from the source message, through the renderers, and to the target message (step 616).

If a send were performed by a local user, the preferred embodiment follows similar steps to those discussed above except that the user information is input from the keyboard 228 or other input device and the user information is sent to the user interface process 306. The user interface process 306 then creates a message from the user information and deposits the created message into the message store 316. In addition, the steps performed by the preferred embodiment for the copy function are similar to the steps performed by the preferred embodiment for the send function except that, for the copy function, the message sink has been predetermined to be the printer.

Figure 7 depicts a flow chart of the steps performed by the preferred embodiment for the receive function. First, the modem 206 receives data from another (sending) modem (step 702). The transport process 324 is then notified that the modem 206 has received data. After being notified, the transport process 324 creates a message, marks the created message as new, and stores the new message in the message store 316 (step 704). Upon finding the new message in the message store 316, the message scheduler 302 determines the message sink for the message (step 706). The message scheduler 302 determines the message sink by accessing the preferences database 318. The message scheduler 302 accesses the preferences database with the destination user name 514 and the preferences database returns an identifier of the preferred message sink for the destination user. The message scheduler 302 then

invokes the format resolution module 314 to create a topology list of renderers and to create a target message (step 708). The format resolution module 314 then determines the capabilities of the message sink by accessing the capabilities database 326 and determines the formatting preferences of the computer user by accessing the preferences database 318 (step 710). Next, the format resolution module 314 creates a topology list of renderers for converting the data from a format acceptable to the modem 206 to a format acceptable to the message sink (step 712). The format resolution module 314 then creates a target message (step 714). After receiving the results of the format resolution module 314, the message scheduler 302 spawns a job process 308, 310 for sending the data from the modem, through the renderers, and to the message sink (step 716).

Figures 8A and 8B depict a flow chart of the steps performed by the message scheduler 302. The message scheduler 302 is invoked upon system startup of the facsimile machine 200 and continues processing until power down of the facsimile machine 200. The message scheduler 302 is responsible for viewing the message store 316, for invoking format resolution 314 for each new message, and for creating a job process 308, 310 for each target message in the message store 316. The message scheduler 302 views the message store 316 upon receiving an interrupt signifying that a new message has arrived or upon the expiration of a predetermined period of time (i.e., polling). The message scheduler 302 first views the message store 316 by reading each message in the message store 316 one at a time (step 802). The message scheduler 302 then determines if the first message encountered is a new message by reading the status field 506 of the encountered message (step 804). If the encountered message is a new message, the message scheduler 302 determines whether the time to send the encountered message has arrived (step 806). Each message in the message store 316 contains a date and time stamp for the delivery of the message (i.e., the delivery time field 510), thus allowing the user to delay the delivery of messages. The message scheduler 302 compares the current date and time to the date and time in the delivery time field 506 for each message to determine whether the time has arrived to send the message.

If the time has arrived to send the encountered message, the message scheduler 302 updates the status information of the encountered message to indicate that the encountered message is schedulable (step 808). If the time has not arrived to send the encountered message or if the encountered message is not a new message, the message scheduler 302 determines whether the encountered message in the message store 316 is schedulable (step 810). If the encountered message is not schedulable, the

message scheduler 302 continues to step 834 in Figure 8B. However, if the encountered message is schedulable, the message scheduler 302 determines if the encountered message is a source message by reading the type field 508 (step 812). If the encountered message is not a source message, the message scheduler 302 determines whether the message source (or sources) and the message sink (or sinks) are available (step 814). If the message source and the message sink are not available, processing continues to step 834 in Figure 8B. However, if the message source and the message sink are available, the message scheduler 302 spawns a job process 308, 310 to process the encountered message (step 816). After the message scheduler 302 has invoked the job process, processing continues to step 834 in Figure 8B. If the encountered message is a source message, the message scheduler 302 determines the message source of the encountered message by reading the message source field 512 of the encountered message (step 818).

The message scheduler 302 next determines if the encountered message is for a receive function (step 820 in Figure 8B). If the encountered message is for a receive function, the message scheduler 302 looks up the preferences of the destination user to determine the message sink (step 822). The message scheduler 302 looks up the preferences of the destination user in the preferences database 318. After the preferences of the destination user have been determined, the message scheduler 302 continues to step 832. However, if the encountered message is not for a receive function, the message scheduler 302 determines if the encountered message is for a send function (step 824). If the message encountered is for a send, the message scheduler 302 looks up the address of the destination user and determines the message sink (step 826). The lookup for the address of the destination user is performed by the message scheduler 302 in the user database 320. After the lookup of the destination user address has completed, the message scheduler 302 continues to step 832. However, if the encountered message is not for a send function, the message scheduler 302 determines whether the encountered message is for a copy function (step 828).

If the encountered message is for a copy function, the message scheduler 302 determines that the printer is the message sink (step 430). After the message scheduler determines that the printer is the message sink, the message scheduler 302 continues to step 832. If the message encountered was not for a copy function, the function of the encountered message has not been determined (step 828) and processing continues with step 834. If the encountered message is for a receive, send, or copy, and the message sink has been determined, the message scheduler 302 invokes the format

resolution module 314 for the encountered message (step 832). The format resolution module 314 is responsible for returning a topology list of renderers for each attachment of the encountered message as well as creating a target message for each message sink of the encountered message. After the format resolution module 314 has completed, the message scheduler 302 determines whether there are more messages in the message store 316 for processing (step 834). If there are more messages in the message store for processing, processing continues with step 802 in Figure 8A (note "C" in Figure 8B) and the message scheduler 302 processes another message (step 834). However, if there are no more messages in the message store 316 for processing, the message scheduler 302 waits until receiving an interrupt or until a predetermined period of time has expired before viewing the message store again (step 836).

Figure 9 depicts a flow chart of the steps performed by the format resolution module 314. In the preferred embodiment of the present invention discussed herein, the format resolution module 314 is a dynamically linked library that is responsible for returning a topology list of renderers for each attachment in a message as well as creating a target message for each message sink of the message. The first step that the format resolution module 314 performs is to determine whether the message sink is a modem 206 (step 902). If the message sink is a modem, the format resolution module 314 determines if the capabilities of the destination modem are in the capabilities database 326 (step 904). The capabilities of a device, such as a modem, refer to the format of the data that is acceptable to that device. An example of a format acceptable to a device is whether the device interprets the first bit in a serially received byte to be the most significant bit or the least significant bit. Another example of a format acceptable to a device is the width of a document. If the capabilities of the destination modem are in the capabilities database 326, the format resolution module 314 looks up the capabilities in the capabilities database 326 for the destination modem (step 910). However, if the capabilities of the destination modem are not located in the capabilities database 326, the format resolution module 314 invokes the local modem 206 to communicate with the destination modem wherein the format resolution module 314 requests the capabilities of the destination modem (step 906). However, if a modem is not the message sink, the format resolution module 314 communicates to the message sink and requests the capabilities of the message sink (step 908).

After the format resolution module 314 has received the capabilities of the message sink, the format resolution module 314 determines the formatting preferences of the user by accessing the preferences database 318 (step 911). After the

formatting preferences of the user have been determined, the format resolution module 314 inputs the capabilities of the message sink, the formatting preferences of the user and the format of the data received from the message source into a preinstalled renderer table to determine a topology of the renderers for converting the data from a format acceptable to the message source into a format acceptable to the message sink. The preinstalled renderer table is a table created off-line which maps the various source formats onto topology lists for a desired destination format. After the topology list for an attachment is created, the format resolution module 314 determines if there are more attachments in the message to be processed (step 914). If there are more attachments in the message to be processed, the format resolution module 314 creates a topology list for another attachment (step 912). However, if no more attachments are contained within the message, the format resolution module 314 creates a target message (step 916). The format resolution module 314 creates the target message with a type field 508 indicating that the message is a target message and with an attachment field containing the file name, pipe or other reference, such as a memory address or a bus address, to the message sink. The format resolution module 314 then determines whether there are more message sinks for the message (step 918). If there are no more message sinks for the message, the format resolution module 314 returns the topology list to the message scheduler 302 and ends processing. However, if there are more message sinks for the message, the format resolution module 314 creates a topology list for each attachment and a target message for the message sink. After each message sink has been processed, the format resolution module 314 returns the topology list to the message scheduler 302 and processing ends.

Figure 10 depicts a flow chart of the steps performed by the job process 308, 310. The job process 308, 310 is responsible for initiating the renderers in a topology list for each attachment and for sending data between the source message, the renderers in the topology list, and to the target message. The job process 308, 310 begins processing by invoking the message sink (step 1002). After the message sink has been invoked, the job process 308, 310 invokes the message source (or sources) (step 1004). Next, the job process 308, 310 creates a pipeline (a path) from the message source through the renderers to the message sink (step 1006). The job process then tells the message source to start sending the data (step 1008). As the message source is sending the data, the job process controls a flow of the data through the renderers and to the message sink (step 1010). After one attachment has been processed and all of the data from the message source for the attachment has been sent to the message sink, the job process determines if there are more attachments to be processed (step 1012). If

there are more attachments to be processed, the job process processes another attachment (step 1012). However, if there are no more attachments to be processed, the job process terminates.

While the present invention has been described with reference to a preferred embodiment thereof, those skilled in the art will know of various changes in form that may be made without departing from the spirit and scope of the claimed invention as defined in the appended claims.

Claims

1. In a communication device for use by a user, the communication device having a message source, a message sink, a database, a format resolution module, and a message scheduler for processing messages, a method for processing messages comprising the steps of:

under the control of the communication device,

receiving user input;

creating a message in response to receiving user input wherein the created message has a message source and a message sink;

storing the message in a message store;

determining when a message has arrived in the message store;

processing new messages in the message store by the message scheduler to create schedulable messages;

when a message is a schedulable message,

spawning a job process by the message scheduler to send data in the message from the message source to the message sink; and

when the message is a source message,

determining the message sink by the message scheduler by accessing a database; and

invoking a format resolution module by the message scheduler to determine a method for converting data within the message from a format acceptable to the message source to a format acceptable to the message sink.

2. The method of claim 1 wherein the step of creating a message includes the step of creating a message in response to user input wherein the created message has a plurality of message sources and a plurality of message sinks.

3. The method of claim 1 wherein each message sink has capabilities for the format of data acceptable to that message sink, wherein the user has preferences, wherein the message has an attachment, and wherein the step of invoking the format resolution includes the steps of:

determining capabilities of the message sink;

determining preferences of the user;

creating a topology list of renderers for each attachment in the message for converting data in the message from a format acceptable to the message source to a format acceptable to the message sink using the preferences of the user; and

creating a target message for each message sink for the message.

4. The method of claim 3 wherein the preferences of a user includes overlay preferences for stylistic information added to a message and branding preferences for a date and time to be printed with a message.

5. The method of claim 3 wherein the communication device has a capabilities database, wherein the step of determining the capabilities of the message sink includes the steps of:

when the message sink is a modem, determining the capabilities of the destination modem by reading the capabilities database;

when reading the capabilities database, if an entry for the destination modem is not found, determining the capabilities of the destination modem; and

when the message sink is not a modem, communicating to the message sink to determine the capabilities of the destination device.

6. The method as recited in claim 1 wherein the communication device has a plurality of renderers for converting data, and wherein the step of spawning a job process includes the steps of :

initializing the message source and message sink to place the message source and the message sink in a running state;

receiving the data from the message source; and

sending the data from the message source, through the renderers to convert the data from a format acceptable to the message source to a format acceptable to the message sink, and to the message sink.

7. The method of claim 1 wherein the communication device is an input and output device.

8. The method of claim 1 wherein the communication device is a facsimile machine.

9. In a facsimile machine, a method for receiving a message comprising the steps of:

under the control of the facsimile machine,

providing a modem that is local to the facsimile machine;

providing a message sink that is local to the facsimile machine;

receiving a message from the modem;

determining preferences of a user of the facsimile machine for the message sink;

determining a method for converting data in the received message into a format acceptable to the message sink; and

spawning a process for sending the data in the received message from the modem to the message sink.

10. The method as recited in claim 9 wherein the step of spawning a process includes the steps of :

initializing the modem and the message sink to place the modem and the message sink in a running state;

receiving data in the message from the modem;

converting the received data from a format acceptable to the modem to a format acceptable to the message sink to create converted data;

formatting the converted data using preferences of the user to create formatted data; and

sending the formatted data to the message sink.

11. The method of claim 9 wherein the facsimile machine has a plurality of renderers, wherein each message has an attachment, wherein the step of determining a method for converting data in the received message includes the steps of:

creating a list of renderers indicating a sequence of renderers in the plurality appropriate for converting the data in the received message from a format acceptable to the modem to a format acceptable to the message sink for each attachment in the received message; and

creating a target message for each message sink of the received message.

12. In a communication machine, the communication machine having a processor, a method for copying a message comprising the steps of:

providing a scanner that is local to the communication machine;

providing a printer that is local to the communication machine;

receiving a message having data from the scanner;

determining a method for converting the data in the received message into a format acceptable to the printer; and

generating a path of execution for the processor for sending the data in the received message from the scanner to the printer.

13. The method as recited in claim 12 wherein the communication device has a plurality of renderers for converting data, wherein the step of generating a path of execution includes the steps of :

initializing the scanner and the printer to place the scanner and the printer in a running state; and

sending the data in the received message from the scanner, through the renderers to convert the data from a format acceptable to the scanner to a format acceptable to the printer, and to the printer.

14. The method of claim 12 wherein the communication device has a plurality of renderers, wherein the step of determining a method for converting data in the received message includes the steps of:

creating a topology list of renderers for converting data in the received message from a format acceptable to the scanner to a format acceptable to the printer; and

creating a target message.

15. In a facsimile machine, the facsimile machine having a message source and a message sink, each having acceptable data formats, a method for sending a message comprising the steps of:

receiving user input into the facsimile machine;

determining the message sink by examining the user input;

determining a method for converting data in the message from a format acceptable to the message source to a format acceptable to the message sink; and

generating a path of execution for sending the data in the message from the message source to the message sink and for converting the data in the message.

16. The method of claim 15 wherein the facsimile machine is connected to a computer, and wherein the step of receiving user input further includes the step of receiving user input via the computer.

17. The method as recited in claim 15 wherein the facsimile machine has a plurality of renderers for converting data, wherein the step of generating a path of execution includes the steps of:

initializing the message source and the message sink to place the message source and the message sink in a running state; and

sending the data in the message from the message source, through the renderers and to the message sink.

18. The method of claim 15 wherein the facsimile machine has a plurality of renderers for converting data, wherein the message has an attachment, wherein the step of determining a method for converting data in the message includes the steps of:

creating a list of renderers for each attachment wherein executing each renderer in the list in an order specified by the list converts the data in the message from a format acceptable to the message source to a format acceptable to the message sink; and

creating a target message for each message sink of the message.

19. The method of claim 15 wherein the facsimile machine has a database, the database having a plurality of user names and a plurality of user addresses, wherein the step of determining the message sink includes the steps of:

selecting a destination user name from the user input; and

determining the message sink by accessing the database with the destination user name.

20. A communication device for processing messages comprising:

a receive component for receiving user input and for creating a message in response to receiving the user input;

a message scheduler for invoking job processes for each message and for invoking a format resolution for each message;

a device for invoking a job process for sending data in a message from a message source to a message sink and for converting the data in the message; and

a device for invoking a format resolution for determining a list of renderers for converting the data in the message from a format acceptable to the message source to a format acceptable to the message sink.

21. A facsimile machine comprising:

a modem for receiving messages;

a message scheduler for invoking a job process for each message received from the modem and for invoking a format resolution for each message;

a job process for sending data in a message from the modem to a message sink;

a format resolution for determining a list of renderers for converting the message from a format acceptable to the modem to a format acceptable to the message sink; and

a message sink for receiving messages from the job process.

22. A facsimile machine for copying messages comprising:

a scanner for inputting documents and creating messages from the documents;

a printer for receiving messages and for printing the received messages;

a processing component for receiving created messages from the scanner and for sending the created messages to the printer, the processing component further including:

a message scheduler for invoking a job process for each message and for invoking a format resolution for each message;

a job process for sending data in the message from the scanner to the printer; and

a format resolution for determining a topology list of renderers for converting the message from a format acceptable to the scanner to a format acceptable to the printer.

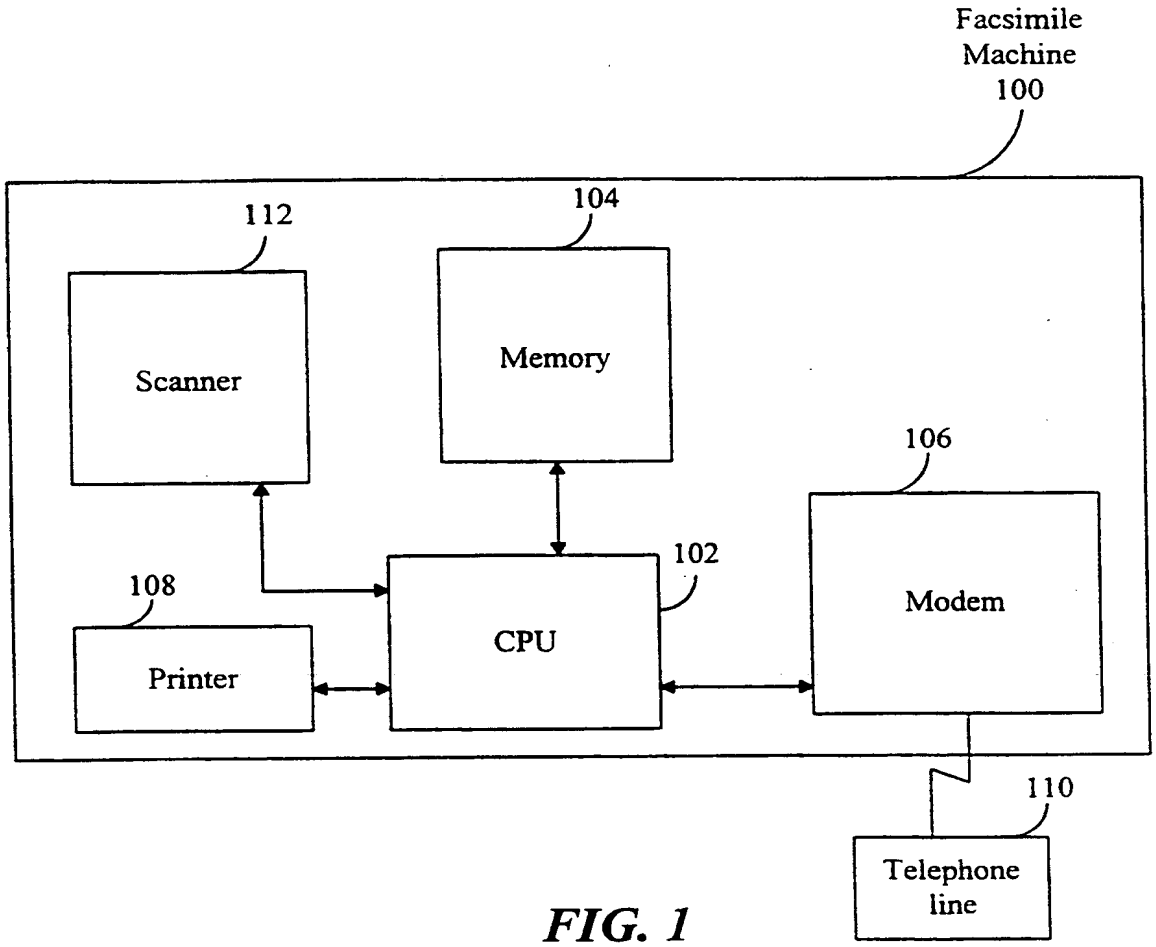


FIG. 1
Prior Art

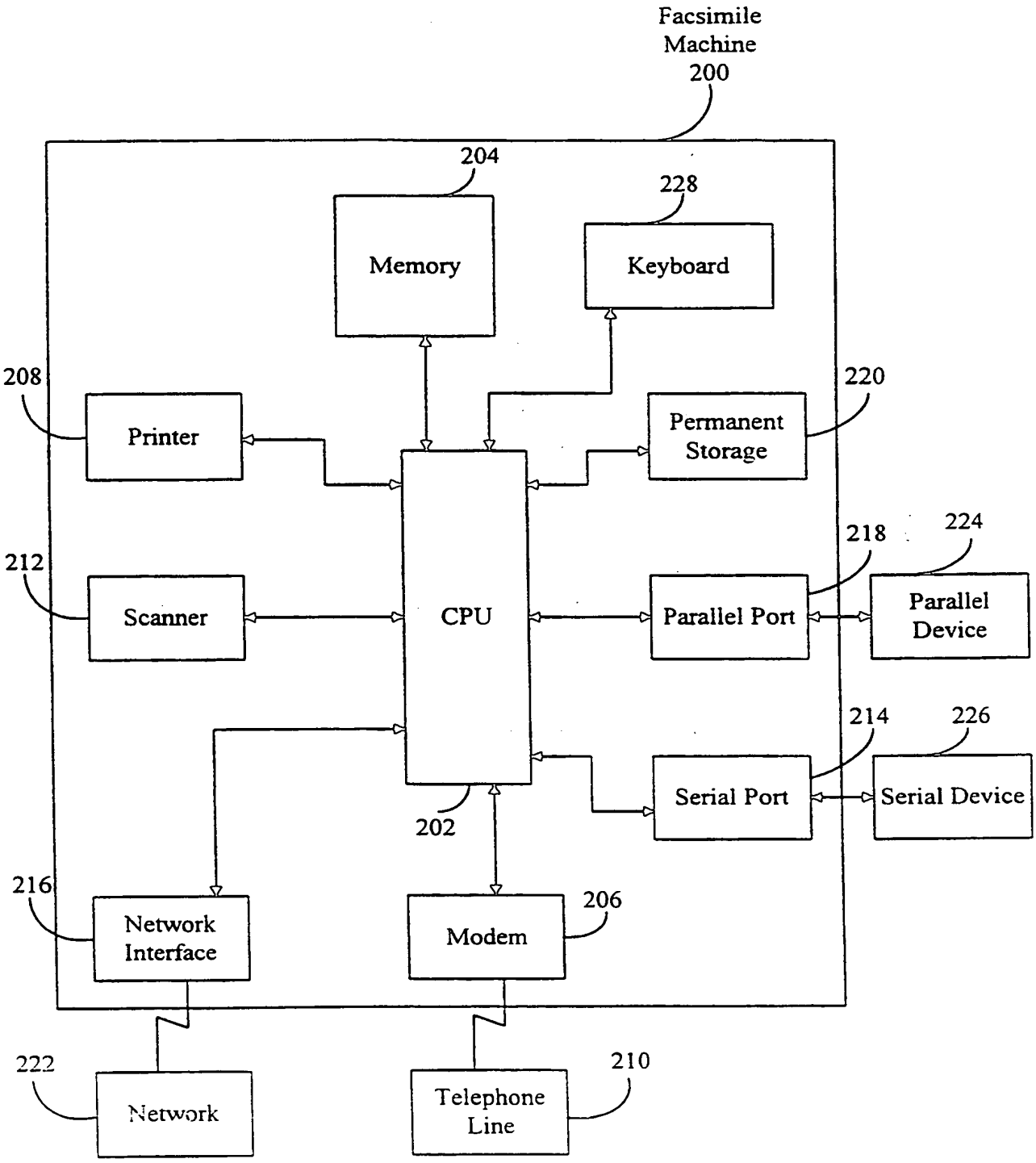

Facsimile Machine 200
204
Memory
228
Keyboard
208
Printer
220
Permanent Storage
212
Scanner
CPU
218
224
Parallel Port
Parallel Device
214
226
Serial Port
Serial Device
202
216
Network Interface
206
Modem
222
Network
210
Telephone Line

FIG. 2

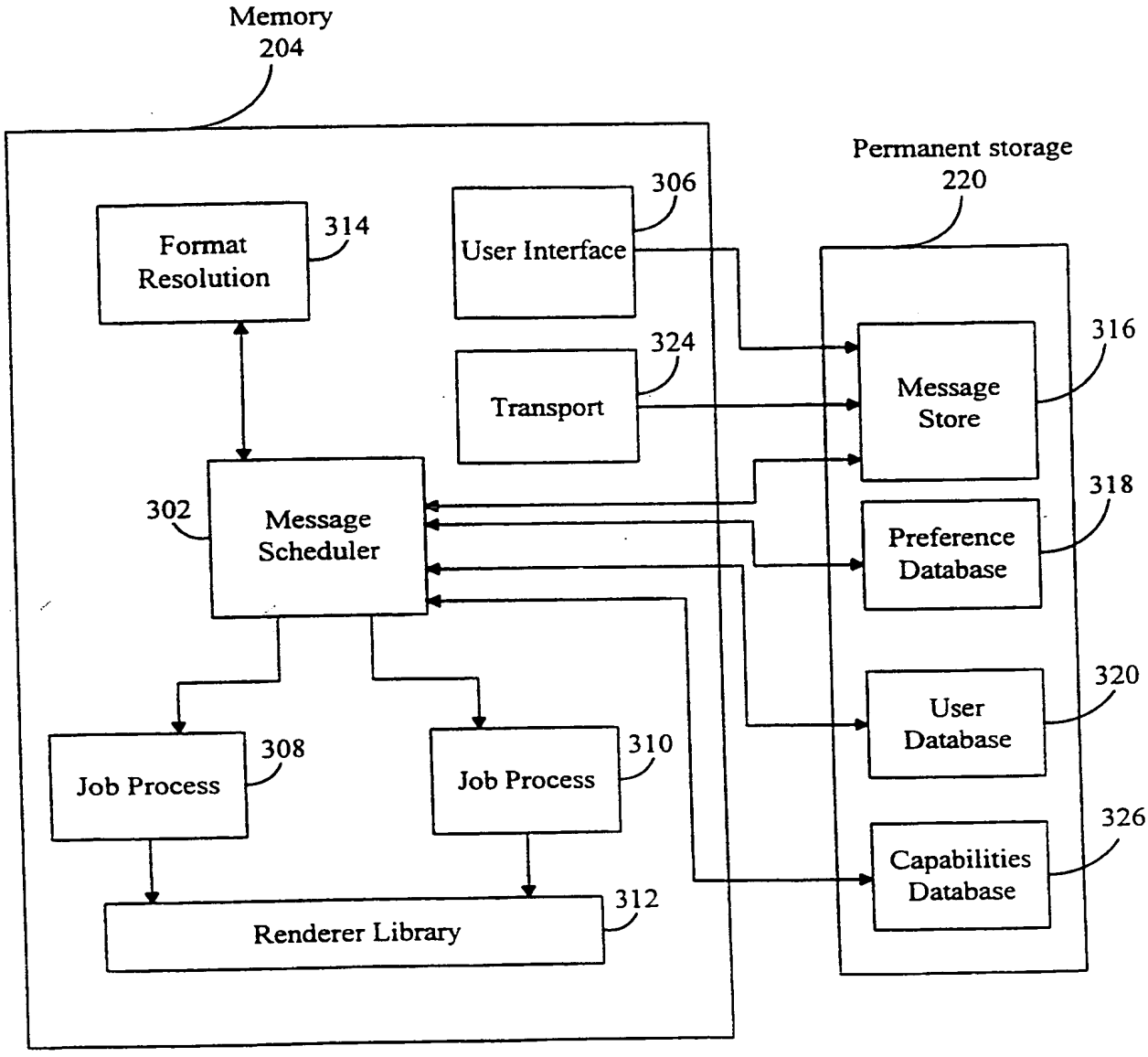

Memory
204
Permanent storage
220
Format Resolution
314
User Interface
306
Transport
324
Message Store
316
Message Scheduler
302
Preference Database
318
User Database
320
Job Process
308
Job Process
310
Capabilities Database
326
Renderer Library
312

FIG. 3

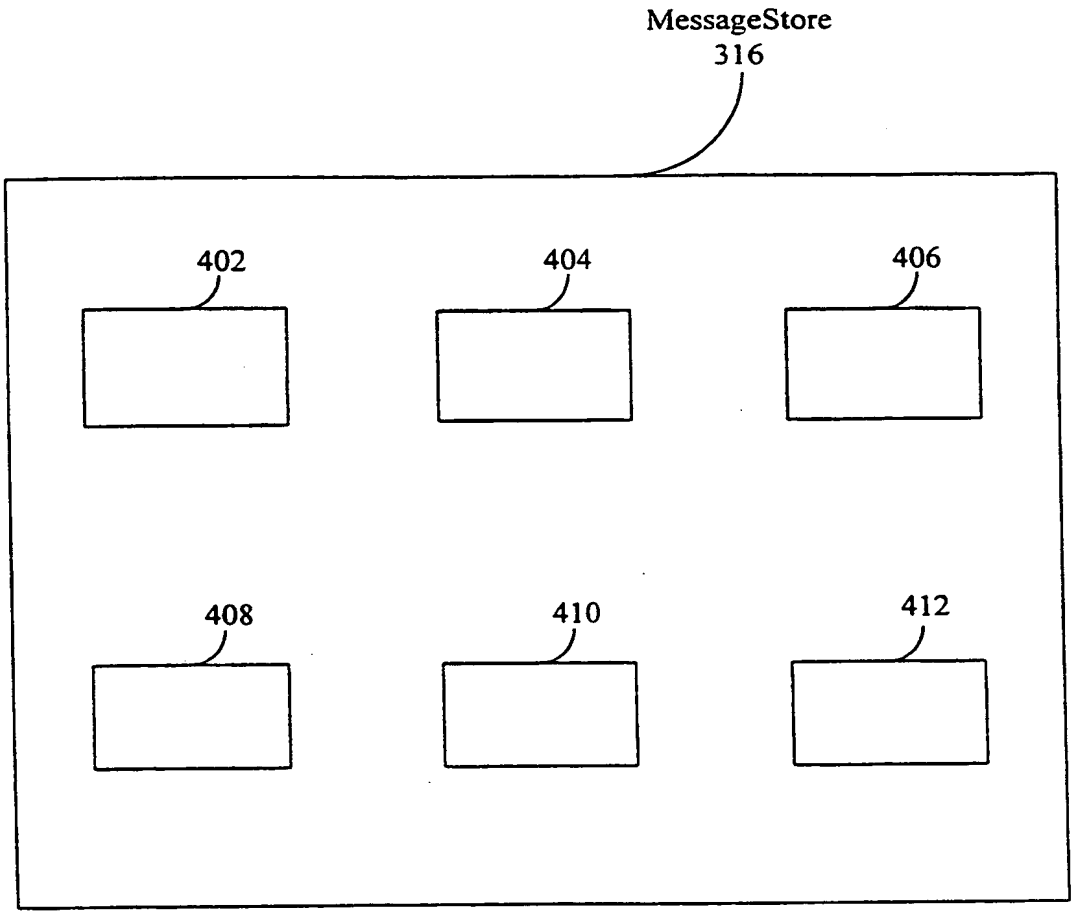


FIG. 4

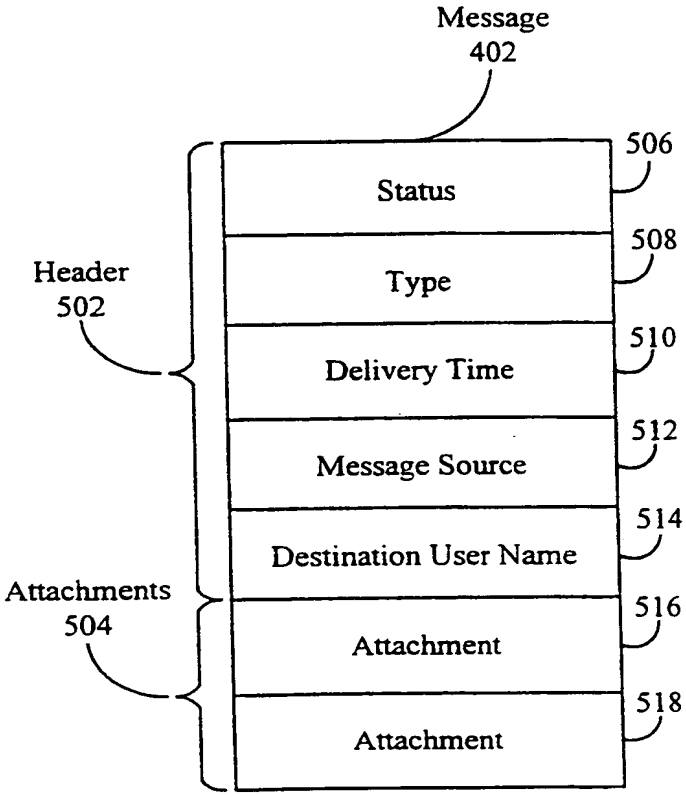

Message
402
Status
506
Type
508
Header
502
Delivery Time
510
Message Source
512
Destination User Name
514
Attachments
504
Attachment
516
Attachment
518

FIG. 5

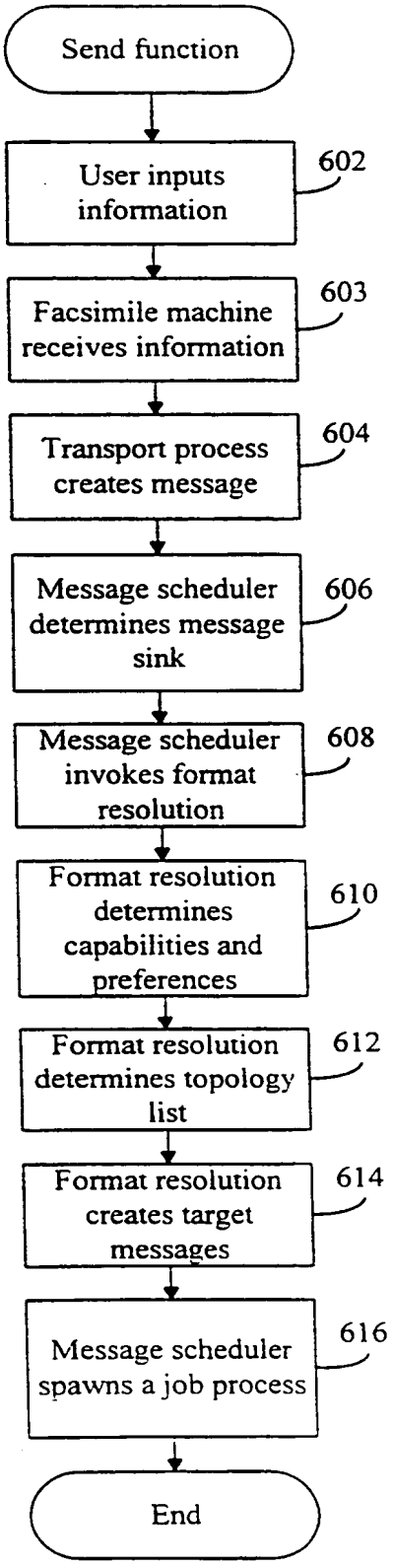

Send function
User inputs information
602
Facsimile machine receives information
603
Transport process creates message
604
Message scheduler determines message sink
606
Message scheduler invokes format resolution
608
Format resolution determines capabilities and preferences
610
Format resolution determines topology list
612
Format resolution creates target messages
614
Message scheduler spawns a job process
616
End

FIG. 6

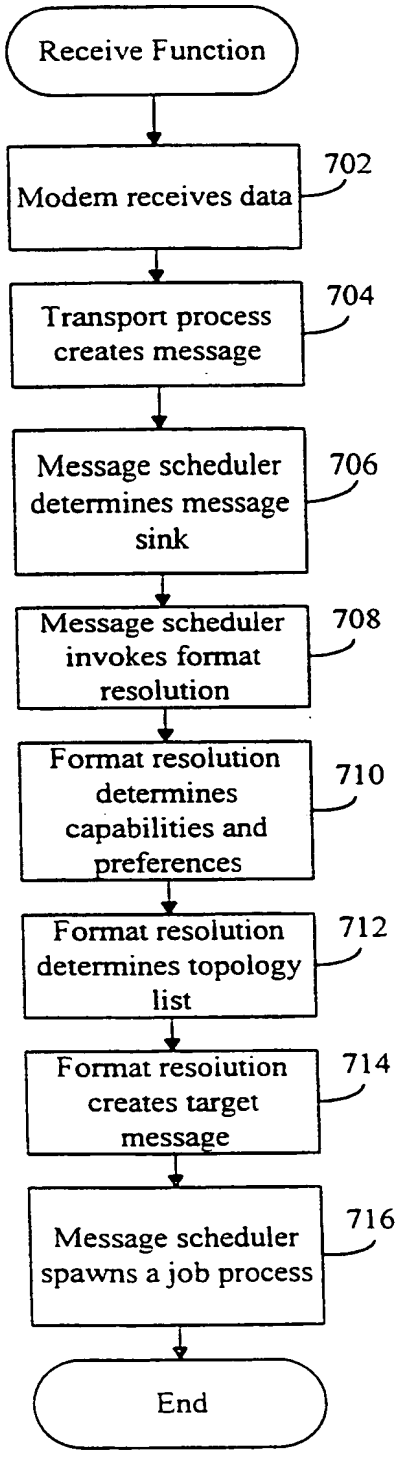

Receive Function
Modem receives data
702
Transport process creates message
704
Message scheduler determines message sink
706
Message scheduler invokes format resolution
708
Format resolution determines capabilities and preferences
710
Format resolution determines topology list
712
Format resolution creates target message
714
Message scheduler spawns a job process
716
End

FIG. 7

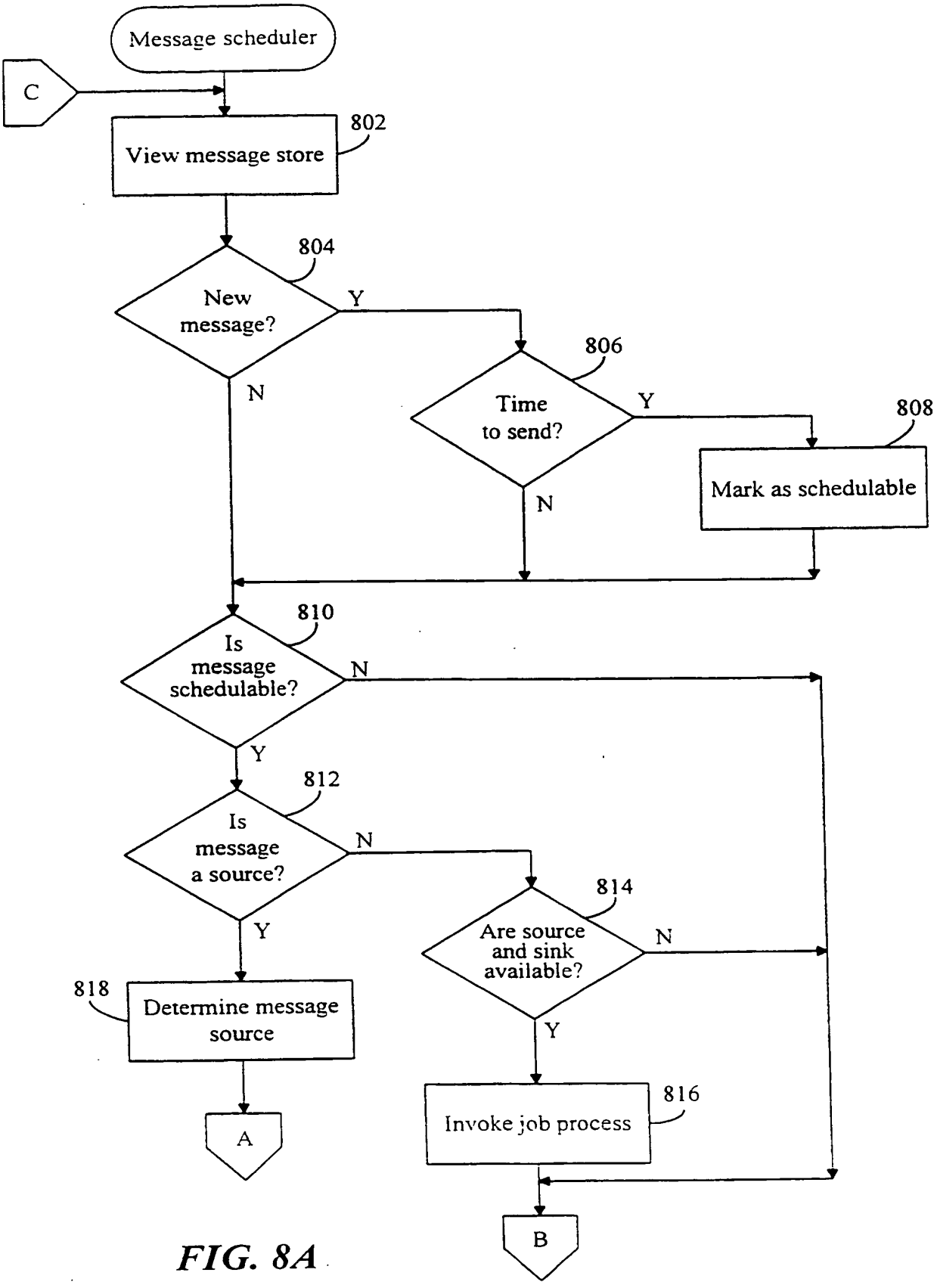

Message scheduler
C
802
View message store
804
New message?
Y
N
806
Time to send?
Y
N
808
Mark as schedulable
810
Is message schedulable?
N
Y
812
Is message a source?
N
Y
814
Are source and sink available?
N
Y
816
Invoke job process
818
Determine message source
A
B

FIG. 8A

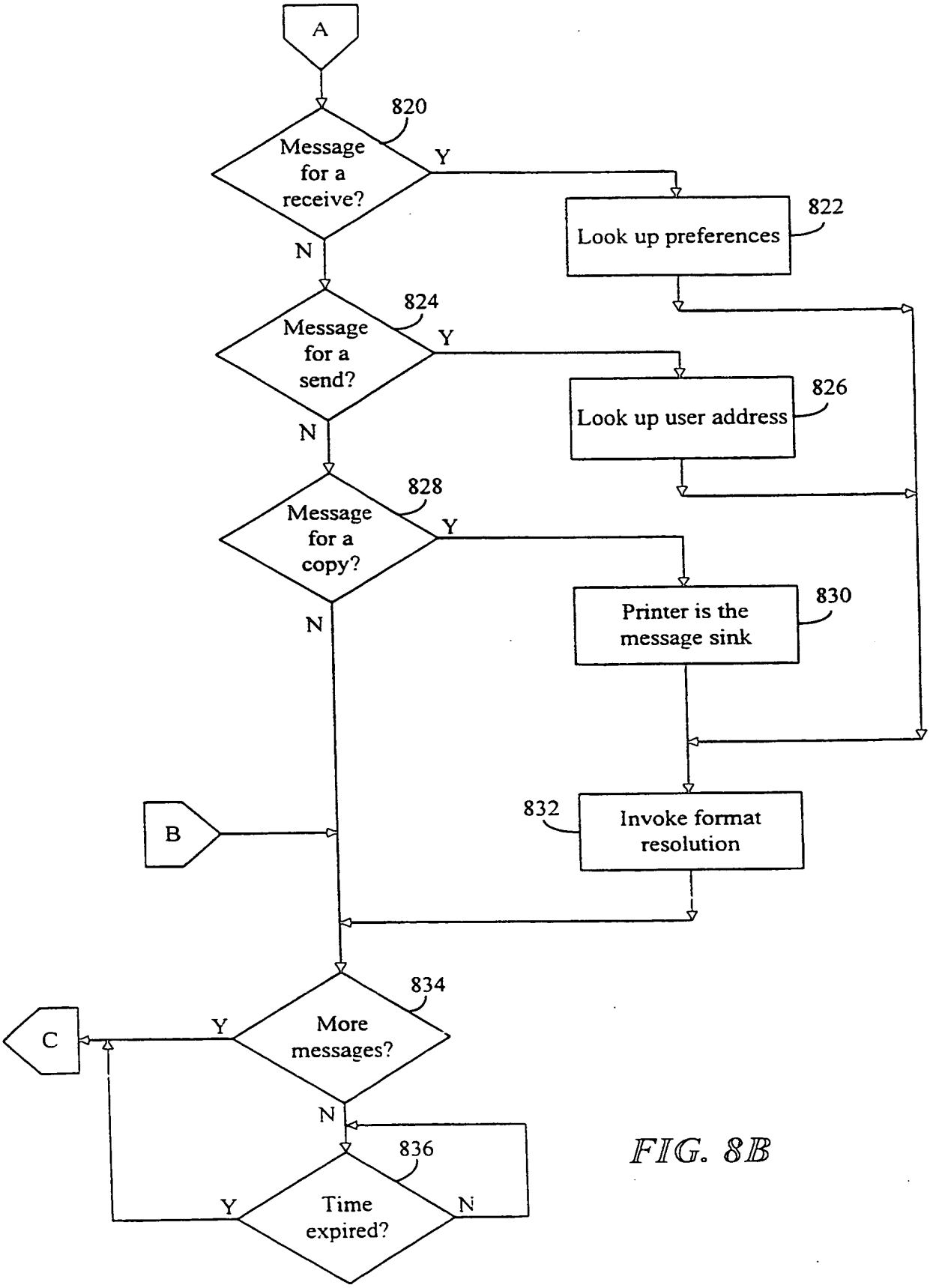

A
820
Message for a receive?
Y
N
822
Look up preferences
824
Message for a send?
Y
N
826
Look up user address
828
Message for a copy?
Y
N
830
Printer is the message sink
B
832
Invoke format resolution
834
More messages?
Y
C
N
836
Time expired?
Y
N

FIG. 8B

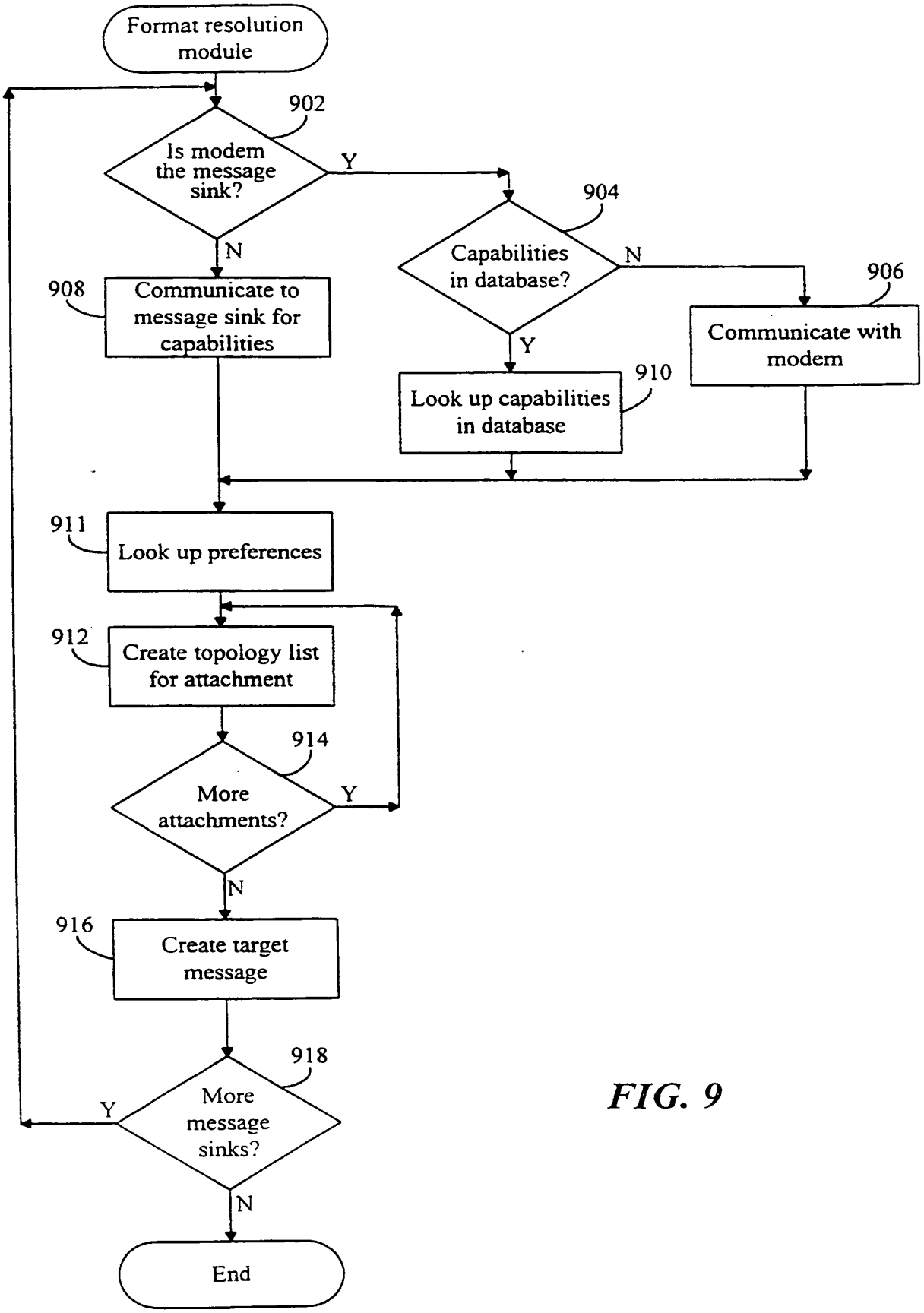

Format resolution module
902
Is modem the message sink?
Y
N
904
Capabilities in database?
N
Y
906
Communicate with modem
908
Communicate to message sink for capabilities
910
Look up capabilities in database
911
Look up preferences
912
Create topology list for attachment
914
More attachments?
Y
N
916
Create target message
918
More message sinks?
Y
N
End

FIG. 9

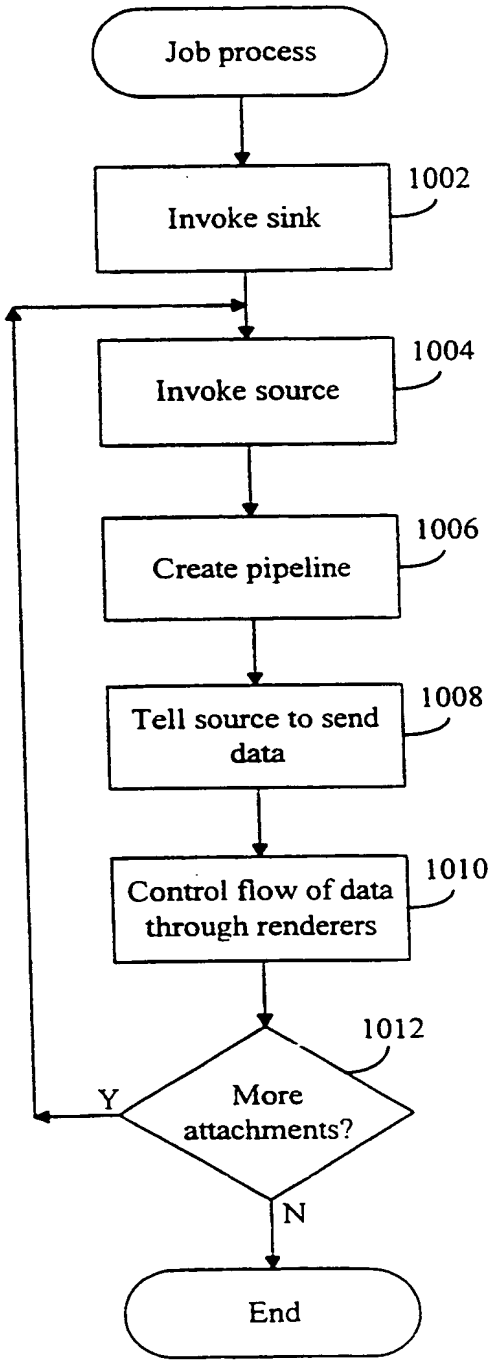

Job process
Invoke sink
1002
Invoke source
1004
Create pipeline
1006
Tell source to send data
1008
Control flow of data through renderers
1010
More attachments?
1012
Y
N
End

FIG. 10

INTERNATIONAL SEARCH REPORT

Interna l Application No
PCT/US 95/02120

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 H04N1/32

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US,A,3 751 582 (WERNIKOFF R ET AL) 7 August 1973 see abstract ---	1,8,9, 15,20-22
A	DE,C,40 10 553 (DIEHL GMBH & CO) 21 November 1991 see abstract ---	1,20
A	EP,A,0 412 461 (SHARP KK) 13 February 1991 see column 7, line 36 - column 9, line 22 ---	1,3,5,8, 9,15,19
A	US,A,5 142 594 (SUGISHIMA KIYOHISA) 25 August 1992 see abstract --- -/--	1,7,12, 20

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
22 June 1995	06.07.95

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+ 31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+ 31-70) 340-3016	Hazel, J

Form PCT ISA 210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Interna ıl Application No
PCT/US 95/02120

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	GB,A,2 154 348 (CITIZEN WATCH CO LTD) 4 September 1985 see page 1, line 5 - line 11	1,7,12, 20
A	PATENT ABSTRACTS OF JAPAN vol. 012 no. 308 (E-647) ,22 August 1988 & JP,A,63 074366 (FUJITSU LTD) 4 April 1988, see abstract	1,8,9, 15,16, 20-22

INTERNATIONAL SEARCH REPORT

Information on patent family members

Interna .l Application No
PCT/US 95/02120

Patent document cited in search report	Publication date	Patent family member(s)		Publication date
US-A-3751582	07-08-73	AU-B-	470860	01-04-76
		AU-A-	4954572	06-06-74
		CA-A-	953814	27-08-74
		DE-A-	2260175	14-06-73
		GB-A-	1420220	07-01-76
		JP-A-	48076422	15-10-73
		JP-B-	52012046	04-04-77
DE-C-4010553	21-11-91	NONE		
EP-A-0412461	13-02-91	JP-A-	3068264	25-03-91
		JP-A-	3068265	25-03-91
		CA-A,C	2022584	08-02-91
US-A-5142594	25-08-92	JP-B-	6093721	16-11-94
		JP-A-	59183574	18-10-84
		US-A-	5162917	10-11-92
GB-A-2154348	04-09-85	JP-C-	1802315	26-11-93
		JP-B-	5008835	03-02-93
		JP-A-	60169893	03-09-85
		DE-A-	3505314	22-08-85
		US-A-	4691364	01-09-87

This Page Blank (uspto)